

8ビット・マイクロプロセッサ：Z80の プログラム開発（II）

福井 稔・岡田俊治・及川浩和

1. は じ め に

前回、中日本自動車短期大学論叢第23号で、8ビット・マイクロプロセッサ：Z80のプログラム開発（I）としてプログラム開発のアセンブル作業の実例までを示しました。今回はその続編として、ROM にプログラムを書き込む作業、テスト作業、デバック作業について実例をまじえながら示します。

2. ROM への書き込み

プログラム開発における HEX ファイルの ROM への書き込みは、ROM ライターが用いられている。市販の ROM ライターには、安価な2～3万のものから30～40万円以上のもので色々な種類がある。我々の所にも3～4種類の ROM ライターがあるが、その中から AVAL 社製の PECKER11 を使用した。PECKER11 は高機能を備えた ROM ライターで、その全てを紹介するには誌面の多くを割かなければならないので、今回は、基本的な ROM への書き込み操作のみを紹介します。

HEX ファイルを書き込むデバイス（ROM）は、EPROM を用いた。EPROM は、紫外線を数分から数時間照射することによって、書き込まれている記録内容を消去し、その後、新たな内容を書き込むことのできる ROM で、ROM ライターを使う前に、EPROM の消去をしておかなければならない。

EPROM の消去にはイレーサーが用いられるが、我々は、電気パーツショップで安価な紫外線殺菌灯（10w）を購入し、ROM の窓に紫外線を照射して消去した。我々の経験では、30分程紫外線を照射すれば消去することができた。以下、PECKER11 の操作方法である。

PECKER11 は、本体のみで使用することもできるが、操作方法が複雑なため、パソコン側からリモートコントロールのできるソフト“AV-Link”（AVAL 製）を使用した。システムは、NEC の PC-9801（または EPSON PC-386, 286）のパソコンと PECKER11 を、専用の RS-232 C インタフェース・ケーブルで接続すればよい。なお、AV-Link のファイルを次に示す。

ドライブ A: のディスクにはボリュームラベルがありません
ディレクトリは A:¥

COMMAND	COM	24161	87-10-23	0:00
AUTOEXEC	BAT	988	88-11-01	1:10
CONFIG	SYS	24	88-11-01	1:10
COMDRV	SYS	3356	88-11-01	1:10
AFDRV	COM	2094	88-11-01	1:10
COMSPED	EXE	14976	88-11-01	1:10
LOGO	EXE	16410	88-11-01	1:10
AVLINK	EXE	74232	88-11-01	1:10
AVLINK	CND	67	88-11-01	1:10
AVLINK	ERR	1171	88-11-01	1:10
RX1	BAT	478	88-11-01	1:10
RX1	MSG	9527	88-11-01	1:10
RX1	MNU	931	88-11-01	1:10
RX1	DEV	6617	88-11-01	1:10
RX1	HLP	1813	88-11-01	1:10
RX2	BAT	478	88-11-01	1:10
RX2	MSG	9865	88-11-01	1:10
RX2	MNU	971	88-11-01	1:10
RX2	DEV	1030	88-11-01	1:10
RX2	HLP	1813	88-11-01	1:10
COND	EXE	26246	88-11-01	1:10
MENU	EXE	19702	88-11-01	1:10
ED	BAT	80	88-11-01	1:10
CONV1	EXE	20030	88-11-01	1:10
TP	EXE	18690	88-11-01	1:10
CONV2	EXE	20470	88-11-01	1:10
READ	ME	1550	88-11-01	1:10
AVLINK	TMP	157712	91-01-17	14:13

28 個のファイルがあります
798720 バイトが使用可能です

① AV-Link を起動する。

A V - L i n k

PC-9801 / MS-DOS 専用
PKW-1100 オンラインソフト
Ver-1.10

Copyright (C) 1988 AVAL CORPORATION

1. PKW-1100とPC-9801をRS-232C
専用ケーブルで接続し、双方の電源を投入します。
2. 次の操作で「AV-Link」を起動します。
A> RX アダプタ番号 ←
3. PKW-1100の次の操作で、メニュー画面が表示
されます。
- SET 0 -又はSET JOB

図2-1 AV-Link 立ち上げ画面

PC-9801と PECKER11 の電源を入れ、すみやかにパソコンのドライブに AV-Link のシステム・ディスクを挿入する。数秒で図2-1のような画面となる。

パソコンのキーボードから、次の下線部を入力する。

RX1

数秒で図2-2のような画面となる。

```

**      「A V - L i n k」オンライン      **                               Ver 1.10

1. 「A V - L i n k」の基本的な動作条件

ディレクトリ          →
ファイル名拡張子      → *
送受信変換            → 無変換
使用ポート            → A U X

2. P K W - 1 1 0 0をオンライン又はターミナルモードにして下さい

操作   :  - S E T  0  - 又は S E T  J O B
    
```

図2-2 パソコンと ROMライターのオンライン設定画面

PECKER11 のキーボードから次のように入力する。

- S E T 0 - J O B

パソコンの画面に図2-3のようなメニューが表示される。

```

選 択 : 「A V - L i n k」-メインメニュー                               Ver-1.10
1: 項目名による P K W - 1 1 0 0 コマンド群からの選択
2: デバイス操作・デバイス関連条件設定コマンド群
3: ファイル送受信・送受信関連条件設定コマンド群
メ ニ ュ ー 4: データ編集・編集関連条件設定コマンド群
5: その他の P K W - 1 1 0 0 コマンド群
6: M S - D O S コマンドの実行
7: その他の「A V - L i n k」コマンド
8: 「A V - L i n k」の終了
実行:
ステータス  |  メーカー名  |  デバイス名  |  ROMサイズ  |  書込方式  |  セットプログラム
              |  NEC          |  2764        |  8KB        |  INTEL-1   |  ALL
              |  マージンチェック  |  RAM先頭    |  転送フォーマット  |  オフセット  |  アタフタ
              |  Enable       |  00000000   |  Intel hex   |  00000000   |  RX01-3.10
結果:
    
```

図2-3 AV-Link メインメニュー画面

以後の操作は、すべて該当するメニュー項目を順に選択すればよい。なお、機能の一端を知っていただく意味で、メインメニューの中の1～4までの各メニュー項目の内容を図2-4(a), (b),

(c), (d)に示す。

1項目名

メニュー	選択: 項目名による P K W - 1 1 0 0 コマンド群からの選択			
	1: 終了	2: セットプログラム	3: ワート*タイプ*	4: マーシチェック
	5: レベルチェック	6: 転送フォーム	7: 編集単位	8: オート*ハイス
	A: テハイス選択	B: ロード	C: オートプログラム	D: フランクチェック
	E: フ*ロク*ラム	F: ヘルファイ	G: EROMイレース	H: ハッファ先頭
	I: テハイス領域	J: I D 表示	K: シリアル送信	L: シリアル受信
	M: セントロ送信	N: オフセット	O: スタート*エント	P: ハッファクリア
	Q: ハッファ反転	R: ハッファ編集	S: フロクム-フ	T: フ*ロクフィル
	U: フロク挿入	V: フロク削除	W: フ*ロクサーチ	X: マニアル操作

図2-4(a) 項目名による P K W - 1100 コマンド群からの選択メニュー画面

デバイス

メニュー	選択: デバイス操作・デバイス関連条件設定コマンド群	
	1: I D コードによる対象デバイスの自動選択	
	2: 対象デバイスの選択	
	3: 対象デバイスからバックアップ R A M へのロード	
	4: プランクチェック・プログラム・ベリファイの連続実行	
	5: その他のデバイス操作コマンド群	
	6: 基本的なデバイス関連条件設定コマンド群	
	7: その他のデバイス関連設定コマンド群	
	8: A V - L i n k メニューへ戻ります	

図2-4(b) デバイス操作・デバイス関連条件設定コマンド群メニュー画面

1ファイル通信

メニュー	選択: ファイル送受信・送受信関連条件設定コマンド群	
	1: R S - 2 3 2 C を使用したファイル送信	
	2: R S - 2 3 2 C を使用したファイル受信	
	3: セントロニクスを使用したファイル送信	
	4: 送受信時のオフセットアドレスの設定	
	5: アスキー H E X のスタート・エンドコードの設定	
	6: P K W - 1 1 0 0 から送信終了時に送信するコードの設定	
	7: データ送受信時の転送フォーマットの選択	
	8: その他のファイル送受信・送受信関連条件設定コマンド群	

図2-4(c) ファイル送受信・送受信関連条件設定コマンド群メニュー画面

1データ編集

メニュー	選択: データ編集・編集関連条件設定コマンド群	
	1: バックアップ R A M 全領域のクリア	
	2: バックアップ R A M 全領域のビット反転	
	3: バックアップ R A M 編集コマンドのマニアル操作	
	4: バックアップ R A M 指定ブロックの指定領域への移動	
	5: 指定データのバックアップ R A M 指定ブロックへの格納	
	6: その他のデータ編集コマンド群	
	7: 編集関連条件設定コマンド群	
	8: A V - L i n k メニューへ戻ります	

図2-4(d) データ編集・編集関連条件設定コマンド群メニュー画面

② ROM に書き込む HEX ファイルを、パソコンから PECKER11 のバッファ RAM へ送信する。メインメニューの“3：ファイル送受信・送受信関連条件設定コマンド群”にカーソルを合わせリターンするとファイル通信のメニューとなる。次にファイル通信のメニューの“RS-232C を使用したファイル送信”にカーソルを合わせリターンすると、図2-5となる。

*** RS-232C 送信ファイルの選択 ***

COMMAND.COM	AUTOEXEC.BAT	CONFIG.SYS	COMDRV.SYS	AFDRV.COM
COMSPED.EXE	LOGO.EXE	AVLINK.EXE	AVLINK.CND	AVLINK.ERR
RX1.BAT	RX1.MSG	RX1.MNU	RX1.DEV	RX1.HLP
RX2.BAT	RX2.MSG	RX2.MNU	RX2.DEV	RX2.HLP
COND.EXE	MENU.EXE	ED.BAT	CONV1.EXE	TP.EXE
CONV2.EXE	READ.ME	AVLINK.TMP	<u>TEST1.HEX</u>	

ファイル名を選択して下さい [

他ドライブ等の場合は、HOME後ファイル名を入力して下さい

図2-5 送信ファイル選択画面

*** RS-232C 送信ファイルの選択 ***

COMMAND.COM	AUTOEXEC.BAT	CONFIG.SYS	COMDRV.SYS	AFDRV.COM
COMSPED.EXE	LOGO.EXE	AVLINK.EXE	AVLINK.CND	AVLINK.ERR
RX1.BAT	RX1.MSG	RX1.MNU	RX1.DEV	RX1.HLP
RX2.BAT	RX2.MSG	RX2.MNU	RX2.DEV	RX2.HLP
COND.EXE	MENU.EXE	ED.BAT	CONV1.EXE	TP.EXE
CONV2.EXE	READ.ME	AVLINK.TMP	TEST1.HEX	

ファイル名を入力して下さい [TEST1.HEX

他ドライブ等の場合は、HOME後ファイル名を入力して下さい

図2-6 ファイル名入力画面

ファイル名の入力を求めて来るので、例えば、ファイル名“TEST1.HEX”にカーソルを合わせリターンすれば図2-6となる。

ファイル名が指定されたところでリターンすると送信が始まる。送信中の画面が図2-7である。(図中の“実行：・・・・”，“結果：・・・・”は前段で別の操作をしていれば、そのときの表示がそのまま残っている) やがて送信が正常に終了すると，“実行：RS-232Cを使用したファイル送信”，“結果：正常に終了しました (OK)”と表示され、図2-8のように自動的にメインメニュー画面に戻る。今回の送信時間は約55秒であった。

ファイル TEST1.HEX 送信中 . . .					
メ ニ ュ -	選択:	RS-232C	を使用したファイル送信		
	1:	RS-232C	を使用したファイル送信		
	2:	RS-232C	を使用したファイル受信		
	3:	セントロニクス	を使用したファイル送信		
	4:	送受信時のオフセットアドレスの設定			
	5:	アスキー-HEXフォーマットのスタート・エンドコードの設定			
	6:	PKW-1100からの送信終了時に送信するコードの設定			
	7:	データ送受信時の転送フォーマットの選択			
ス テ - タ ス	実行:	その他のファイル送受信・送受信関連条件設定コマンド群			
	メーカ名	デバイス名	ROMサイズ	書込方式	セットプログラム
	NEC	2764	8KB	INTEL-1	All
	マシニングチェック	RAM先頭	転送フォーマット	オフセット	アドレス
	Enable	00000000	Intel hex	00000000	RX01-3.10
	結果:				

図2-7 ファイル送信画面

(シリアル送信) 実行結果: 正常に終了しました (OK)					
メ ニ ュ -	選択:	『AV-Link』メインメニュー Ver-1.10			
	1:	項目名によるPKW-1100コマンド群からの選択			
	2:	デバイス操作・デバイス関連条件設定コマンド群			
	3:	ファイル送受信・送受信関連条件設定コマンド群			
	4:	データ編集・編集関連条件設定コマンド群			
	5:	その他のPKW-1100コマンド群			
	6:	MS-DOSコマンドの実行			
	7:	その他の『AV-Link』コマンド			
ス テ - タ ス	実行:	RS-232Cを使用したファイル送信			
	メーカ名	デバイス名	ROMサイズ	書込方式	セットプログラム
	NEC	2764	8KB	INTEL-1	All
	マシニングチェック	RAM先頭	転送フォーマット	オフセット	アドレス
	Enable	00000000	Intel hex	00000000	RX01-3.10
	結果:	正常に終了しました (OK)			

図2-8 ファイル送信終了画面

③ デバイス (ROM) を選択する。メインメニューから“2: デバイス操作・デバイス関連条件設定コマンド群”にカーソルを合わせリターンすると、デバイスに関するメニュー画面となる。次に“2: 対象デバイスの選択”にカーソルを合わせてリターンすると図2-9のような画面となる。

** デバイスの選択 **

A U T O G I N S SGS-THOMSON W S I	A M D 日立 <div style="border: 1px solid black; padding: 2px;">N E C</div> シャープ X I C O R	A T M E L インテル 沖 T E X A S 不明	E X E L 松下 リコー 東芝	富士通 三菱 S E E Q V T I
---	---	---	----------------------------	-------------------------------

	メーカー名 メーカー名 を選択して下さい		ROM容量		書き込み方式
メーカー名	デバイス名				

図2-9 デバイス・メーカー選択画面

以下、この場合用いた EPROM は、NEC 製の“D 2764”であるので、メーカー名：NEC，リターンで図2-10，デバイス名：2764，リターンで図2-11，書き込み方式：INTEL-1，リターンで図2-12となる。そしてもう一度リターンすると，ROM 選択が正常に終了すれば，“実行：対象デバイスの選択”，“結果：正常に終了しました (OK)”と図2-13のように表示され，書き

** デバイスの選択 **

<div style="border: 1px solid black; padding: 2px;">2764</div> 27C256 27C1024	27128 27C256A 28C64	27256 27C512 27C1000A	27256A 27C1000 27C1001A	27C64 27C1001 27C2001
---	---------------------------	-----------------------------	-------------------------------	-----------------------------

	デバイス名 を選択して下さい		ROM容量		書き込み方式
メーカー名	デバイス名				
N E C					

図2-10 デバイス名選択画面

込む ROM の種類が選択できる。ROM 容量は自動的に設定されるようになっている。

** デバイスの選択 **
 INTEL-1 INTEL-2 不明

 メーカー名 書き込み方式 ROM容量 書き込み方式
 NEC 2764

図 2-11 書込み方式選択画面

```

** デバイスの選択 **


INTEL-1          INTEL-2          不明



メーカー名      デバイス名      ROM容量      書込み方式
NEC              2764                  INTEL-1
```

図 2-12 デバイス選択設定画面

1 デバイス	選択：デバイス操作・デバイス関連条件設定コマンド群				
	1: I Dコードによる対象デバイスの自動選択				
	2: 対象デバイスの選択				
メニュー	3: 対象デバイスからバッファRAMへのロード				
	4: ブランクチェック・プログラム・ベリファイの連続実行				
	5: その他のデバイス操作コマンド群				
	6: 基本的なデバイス関連条件設定コマンド群				
	7: その他のデバイス関連設定コマンド群				
	8: A V - L i n kメニューへ戻ります				
	実行：対象デバイスの選択				
	メーカー名	デバイス名	ROMサイズ	書込方式	セットプログラム
	NEC	2764	8KB	INTEL-1	All
ステータス	マージンチェック RAM 先頭 転送フォーマット オフセット アタクタ				
	Enable	00000000	Intel hex	00000000	RX01-3.10
	結果：正常に終了しました (OK)				

図 2-13 デバイス選択終了画面

④ HEX ファイルを送信した PECKER11 のバッファ ROM から、HEX ファイルを EPROM へ書き込む。デバイスに関するメニューの“4：ブランクチェック・プログラム・ベリファイの連続実行”にカーソルを合わせリターンすると、ほんの数秒で、RAM のブランク（消去）チェックを行い、続いて書き込み、そしてベリファイ（書き込んだ内容とバッファ ROM の内容を照合）を一気に行う。書き込みが完了すれば“実行：ブランクチェック・プログラム・ベリファイの連続実行”，“結果：正常に終了しました (OK)”と図 2-14 のように表示される。

1 デバイス	選択：デバイス操作・デバイス関連条件設定コマンド群				
	1: I Dコードによる対象デバイスの自動選択				
	2: 対象デバイスの選択				
メニュー	3: 対象デバイスからバッファRAMへのロード				
	4: ブランクチェック・プログラム・ベリファイの連続実行				
	5: その他のデバイス操作コマンド群				
	6: 基本的なデバイス関連条件設定コマンド群				
	7: その他のデバイス関連設定コマンド群				
	8: A V - L i n kメニューへ戻ります				
	実行：ブランクチェック・プログラム・ベリファイの連続実行				
	メーカー名	デバイス名	ROMサイズ	書込方式	セットプログラム
	NEC	2764	8KB	INTEL-1	All
ステータス	マージンチェック RAM 先頭 転送フォーマット オフセット アタクタ				
	Enable	00000000	Intel hex	00000000	RX01-3.10
	結果：正常に終了しました (OK)				

図 2-14 デバイスへの書き込み終了画面

3. ディスアセンブル

前章までは、Z80のマイコン・プログラムを ROM に書き込むまでの通常のプログラム開発の

手順を追ってきたが、ときにはこの逆の過程を経たいときがある。すなわち ROM 中の機械語プログラムを ROM ライターで読み出し、ニモニック・コードへ逆変換して、プログラムの解説を行なう。この HEX 又は BIN ファイルをニモニックのソース・ファイルへ変換するソフトウェアをディスアセンブラという。つぎに、そのようなディスアセンブラ ZZ (マイクロオーグ) の実行例を示す。

3. 1. 逆アセンブラ ZZ

ZZ はスクリーン・エディタタイプのクロス逆アセンブラソフトで、HEX 型式や BIN 型式のプログラムからアセンブラ・ソースプログラムを生成することができる。したがって、プログラム解析など、リバース・エンジニアリングへの利用が可能である。ターゲット CPU として Z80 の他に、64180, 8085, 8048, 8049, 8031, 8051, 6502 を対象としている。

HEX 型式や BIN 型式のプログラムから完全なアセンブラ・ソースプログラムを自動的に生成できれば理想的であるが、実際にはコード中にバイトデータやワードデータが含まれていることが多く、出力されたコードが、実際はコードであるのかデータであるのかを、すべて自動的に判断して逆アセンブルすることは事実上困難である。したがって、使用エリア以外へのジャンプやコール、リターン命令、ならびに同一のコードが連続しているものなどがあれば、それを手がかりに、どの部分がコードエリアであるのかデータエリアであるのか、あるいはプログラムエリアであるのか、などのことを解析してコードの属性を決定していく必要がある。しかし、簡潔なプログラムならともかく、複雑なものや、プログラム保持のため任意に加工されたコードの場合、解析するのは至難の業で、解析するよりはむしろ全く初めからアセンブラ・ソースプログラムを作成した方が早いとも言われている。

3. 2. ZZ の実行例

ZZ には、ROM からコードを読み出すプログラムが用意されていないため、BASIC で作成した。(図 3-1)

```

10 'MAIN PROGRAM ROM からノ ユニタシ & HEXデータ ノ サクセイ
20 DIM W$(127,15),V$(127),P$(127),Q$(127)
30 INPUT "ROM Type No.=";R$
40 INPUT "FILE NAME for ZZ=";N$
50 GOSUB *SRR
60 INPUT "First address=";F$
70 INPUT "Last address=";L$
80 GOSUB *SDH
90 GOSUB *SCH
100 END
110 'SUBROUTINE ROM からノユニタシ
120 *SRR
130 OPEN "COM1:N81XS" AS #1
140 PRINT #1,"T";R$
150 GOSUB *TIMER
160 PRINT #1,"B"
170 GOSUB *TIMER
180 PRINT #1,"L"
190 GOSUB *TIMER
200 PRINT #1,"D00,07FF"
210 GOSUB *TIMER

```

```

220 FOR I=1 TO 8:INPUT #1,A$:NEXT I
230 PRINT "No.":" ":"ADR  0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F"
240 'LPRINT "No.":" ":"ADR  0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F"
250 FOR I=0 TO 127
260 INPUT #1,A$
270 FOR J=0 TO 15
280 W$(I,J)=MID$(A$,3*J+6,2)
290 NEXT J
300 PRINT I:" ":"A$
310 'LPRINT I:" ":"A$
320 NEXT I
330 CLOSE
340 RETURN
350 END
360 'SUBROUTINE タイマ-
370 *TIMER
380 FOR I=1 TO 10000:NEXT I
390 RETURN
400 END
410 'SUBROUTINE コートノ ヘンカン
420 *SDH
430 I=VAL("&H"+MID$(F$,2,2))
440 J=VAL("&H"+RIGHT$(F$,1))
450 IE=VAL("&H"+MID$(L$,2,2))
460 JE=VAL("&H"+RIGHT$(L$,1))
470 K=0:C=0
480 C=C+1
490 V$(K)=V$(K)+W$(I,J)
500 IF C=16 THEN GOSUB *SDH1
510 J=J+1
520 IF J=16 THEN I=I+1:J=0
530 IF I=IE AND J>JE THEN GOSUB *SDH2:RETURN
540 GOTO 480
550 END
560 'SUBROUTINE
570 *SDH1
580 GOSUB *SDH2
590 K=K+1:C=0
600 F$=HEX$(VAL("&H"+F$)+16)
610 RETURN
620 END
630 'SUBROUTINE
640 *SDH2
650 P$(K)=RIGHT$("0"+HEX$(LEN(V$(K))/2),2)
660 Q$(K)=RIGHT$("0000"+F$,4)
670 RETURN
680 END
690 'SUBROUTINE チェックサム ラ ツケル
700 *SCH
710 OPEN N$ FOR OUTPUT AS #2
720 FOR I=0 TO K
730 G=0
740 B$=":"+P$(I)+Q$(I)+"00"+V$(I)
750 FOR J=2 TO LEN(B$) STEP 2
760 G=G+VAL("&H"+MID$(B$,J,2))
770 NEXT J
780 CS=G
790 IF RIGHT$("0"+HEX$(G+CS),2)="00" THEN 820
800 CS=CS+1
810 GOTO 790
820 PRINT #2,B$:RIGHT$("0"+HEX$(CS),2)
830 NEXT I
840 PRINT #2,":00000001FF"
850 CLOSE
860 RETURN
870 END

```

図3-1 読み出し変換プログラム

読み出しに使用した機種は P-ROM ライタ (PZ-W2) で本来は ROM にプログラムを書込むためのものであるが、オプションにメモリのダンプ表示機能があり、そのダンプデータを HEX 型式データに変換して ZZ で逆アセンブルを行う。ROM の内容は、前回実例として用いた LED 点滅プログラムである。

このプログラムは、バイトデータとワードデータがないなど、比較的コードの追跡が容易なプログラムでは複雑な解析を必要としないが、一般に、ROM からより完全なアセンブラ・ソースプログラムを生成するには、解析作業にかなりの時間と労力を要することになる。次に具体的な手順を示す。

初めに、自動モードでのラベルの生成、属性の決定を行う。属性の変更が必要であればエリアの設定を行う。次に、ラベルやコメントをセットして ZZ を終了する。こうした解析作業に必要な操作は、ファンクション・キーを選択して行うことができる。ファンクション・キーによる操作一覧表を図 3-2 に示す。終了と同時に全エリアの属性と、ラベル、クロスリファレンス、タグなどの情報もすべてファイルにセーブされる。つまり、コードの属性を解析したものをすべてファイルにセーブし、そのファイルの情報を参照して逆アセンブルが行われる。

* ファンクション・キーによる操作一覧表 *

F 1	F 2	F 3	F 4	F 5
ZZ の 終了	ファイル 名の変更	逆アセンブル の結果書込み	コメントの セット、削除	ラベルの セット、削除

ユーザ・タグ ジャンプ	サーチ・タグ ジャンプ	クロス・タグ ジャンプ	×	画面の リカバー
S F 6	S F 7	S F 8	S F 9	S F 10
F 6	F 7	F 8	F 9	F 10
ユーザ・タグ セット、削除	サーチ	クロス・リ ファレンス	インフォ メーション	ビルドと 条件設定

図 3-2 ファンクション・キーによる操作一覧表

次に具体的な手順を示す。

- ① ROM の内容を読み出すために EP-ROM2716 を P-ROM ライタに装着して、読み出し変換プログラム (図 3-1) を実行する。
- ② ROM のタイプを入力する。実例では EP-ROM2716 を使用したので 2716 と入力する。対象 ROM として EP-ROM2716, 27128, 27128A, 27128F, 27128N, 27256, 27256F, 2732, 2764, 2764F, 2764N, EEP-ROM2816A, 2864A がある。
- ③ HEX 型式データをセーブするファイル名を入力する。ファイル名は、FTEST1. HEX とした。
- ④ ダンプデータが図 3-3 のようにプリントアウトされるので、インストラクション・コードとして必要な範囲を入力する。図 3-3 では、010A 番地～012E 番地までにコードがあるので、010A, 012E を入力する。

No.	ADR	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0000	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
1	0010	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
2	0020	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
3	0030	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
4	0040	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
5	0050	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
6	0060	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
7	0070	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
8	0080	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
9	0090	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
10	00A0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
11	00B0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
12	00C0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
13	00D0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
14	00E0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
15	00F0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
16	0100	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	3E	90	D3	03	31	00
17	0110	87	3E	00	D3	02	CD	22	01	3E	FF	D3	02	CD	22	01	C3
18	0120	11	01	16	E7	1E	5F	1D	C2	26	01	15	C2	24	01	C9	FF
19	0130	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
20	0140	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
21	0150	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
22	0160	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
23	0170	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
24	0180	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
25	0190	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
26	01A0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
27	01B0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
28	01C0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
29	01D0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
30	01E0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
31	01F0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
32	0200	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

図3-3 ダンプデータ

- ⑤ 図3-4のような HEX型式データがファイルされる。

```
A>TYPE FTEST1.HEX
:10010A003E90D3033100873E00D302CD22013EFF49
:10011A00D302CD2201C3110116E71E5F1DC22601BB
:05012A0015C22401C90B
:00000001FF
```

図3-4 HEX 型式データ

- ⑥ エディタで図3-5のようなターゲット CPU とターゲットプログラムを指定するプロジェクト・ファイルを作成する。ファイル名は、FTEST1. PRJ とした。

```
A>TYPE FTEST1.PRJ
Z80
HEX_NAME FTEST1.HEX
```

図3-5 プロジェクトファイル

⑦ ZZ のシステムディスクをパソコンのドライブAに入れて ZZ を実行する。ZZ のシステムディスクの内容を図3-6に示す。ZZ を実行するには、

A > ZZ FTEST1. PRJ

と入力する。

⑧ 逆アセンブルが成功すると、図3-7のような画面が表示される。

⑨ 解析作業として f・10 キーを押してビルドと条件設定を選択する。ビルドとは、属性のセットとラベルを作ることをいう。

次に f・4, f・5 キーを押して、未定義ラベルやコメントをセットする。(図3-8)

⑩ f・1 キーを押して ZZ を終了する。終了と同時に、逆アセンブルしたファイル(FTEST1. MAC), 定数定義ファイル (FTEST1. EQU), マップファイル (FTEST1. MAP), ラベルファイル (FTEST1. LBL), コメントファイル (FTEST. CMT), タグファイル (FTEST1. ZTG) がセーブされる。(図3-9) FTEST1. MAC, FTEST1. LBL, FTEST1. MAP の内容を図3-10(a), (b), (c)に示す。マップファイルの“I”はInstructionのIを表し, “I”は1バイト命令, “I 1”は2バイト命令, “ILC”は3バイト命令を表している。また, ラベルがある場合, “I 1”は“i 1”に, “ILC”は“I 1 C”になる。また, “I”の他には“B”バイトデータ, “W”ワードデータがある。

ドライブ A: のディスクのボリュームラベルはありません。
ディレクトリは A:¥ZZ

.		<DIR>	92-01-31	9:57
..		<DIR>	92-01-31	9:57
INSTALL	BAT	475	90-08-05	16:37
README	DOC	1082	90-08-05	16:44
UPDATE	DOC	1835	90-08-05	16:25
ZZ	EXE	135874	90-08-05	16:33
TEST2	HEX	4665	89-11-22	9:57
ZZ0	HLP	1033	89-11-26	21:54
ZZ6502	HLP	258	90-08-05	10:57
ZZ8048	HLP	542	90-08-05	10:57
ZZ8051	HLP	378	89-11-26	21:54
ZZ8085	HLP	196	90-08-05	10:57
ZZCTRL	HLP	973	90-08-05	10:57
ZZF1	HLP	433	90-08-05	10:57
ZZF10	HLP	718	90-08-05	10:57
ZZF2	HLP	945	90-08-05	10:57
ZZF3	HLP	939	90-08-05	10:57
ZZF4	HLP	273	90-08-05	10:57
ZZF5	HLP	1168	90-08-05	10:57
ZZF6	HLP	658	90-08-05	10:57
ZZF7	HLP	1092	90-08-05	10:57
ZZF8	HLP	1011	90-08-05	11:06
ZZF9	HLP	106	90-08-05	10:57
64180	TBL	21467	90-04-09	19:58
I8048	TBL	5835	90-04-09	20:14
I8051	TBL	6951	90-04-09	20:14
I8085	TBL	5555	90-04-09	20:14
M6502	TBL	6266	90-04-09	20:14
Z80	TBL	20459	90-04-09	20:14

29 個のファイルがあります。
904192 バイトが使用可能です。

図3-6 ZZシステムディスクの内容

```

Disassembler for Z80 et'c ZZ.EXE Ver2.06 (c)Micro-org 1989 FTEST1.PRJ Z80
X 010A      LD      A,090H          ;3E 90      >
X 010C      OUT     (003H),A        ;D3 03      ..
X 010E      LD      SP,08700H       ;31 00 87    1..
X 0111      LD      A,000H          ;3E 00      >
X 0113      OUT     (002H),A        ;D3 02      ..
X 0115      CALL    UD_0122         ;CD 22 01    ..
X 0118      LD      A,0FFH          ;3E FF      >
X 011A      OUT     (002H),A        ;D3 02      ..
X 011C      CALL    UD_0122         ;CD 22 01    ..
X 011F      JP      UD_0111         ;C3 11 01    ...
X 0122      LD      D,0E7H          ;16 E7      ..
X 0124      LD      E,05FH          ;1E 5F      ..
X 0126      DEC     E               ;1D         ..
X 0127      JP      NZ,UD_0126      ;C2 26 01    .&.
X 012A      DEC     D               ;15         ..
X 012B      JP      NZ,UD_0124      ;C2 24 01    .$.
X 012E      RET                     ;C9         .

```

END FILE WRITE COMENT LABEL USER T SEARCH X FER MASK BUILD

図3-7 ZZ実行画面1

```

Disassembler for Z80 et'c ZZ.EXE Ver2.06 (c)Micro-org 1989 FTEST1.PRJ Z80
010A      ;メインプログラム
I 010A      LD      A,090H          ;3E 90      >
I 010C      OUT     (003H),A        ;D3 03      ..
I 010E      LD      SP,08700H       ;31 00 87    1..
I 0111 J1:   LD      A,000H          ;3E 00      >
I 0113      OUT     (002H),A        ;D3 02      ..
I 0115      CALL    TIMER          ;CD 22 01    ..
I 0118      LD      A,0FFH          ;3E FF      >
I 011A      OUT     (002H),A        ;D3 02      ..
I 011C      CALL    TIMER          ;CD 22 01    ..
I 011F      JP      J1              ;C3 11 01    ...
0122      ;サブルーチン
I 0122 TIMER: LD      D,0E7H          ;16 E7      ..
I 0124 J2:   LD      E,05FH          ;1E 5F      ..
I 0126 J3:   DEC     E               ;1D         ..
I 0127      JP      NZ,J3           ;C2 26 01    .&.
I 012A      DEC     D               ;15         ..
I 012B      JP      NZ,J2           ;C2 24 01    .$.
I 012E      RET                     ;C9         .

```

END FILE WRITE COMENT LABEL USER T SEARCH X FER MASK BUILD

図3-8 ZZ実行画面2

DIR FTEST1.*

ドライブ A: のディスクのボリュームラベルはありません。
ディレクトリは A:¥

FTEST1	HEX	127	92-09-22	16:18
FTEST1	PRJ	26	92-09-17	15:23
FTEST1	MAC	440	92-09-22	16:40
FTEST1	EQU	0	92-09-22	16:40
FTEST1	MAP	65536	92-09-22	16:40
FTEST1	LBL	47	92-09-22	16:40
FTEST1	CMT	170	92-09-22	16:40
FTEST1	ZTG	178	92-09-22	16:40

8 個のファイルがあります。
832512 バイトが使用可能です。

図3-9 ZZ生産ファイル

A>TYPE FTEST1.MAC

;メインプログラム

	LD	A,090H	;3E 90	>
	OUT	(003H),A	;D3 03	..
	LD	SP,08700H	;31 00 87	1..
J1:	LD	A,000H	;3E 00	>
	OUT	(002H),A	;D3 02	..
	CALL	TIMER	;CD 22 01	..
	LD	A,0FFH	;3E FF	>
	OUT	(002H),A	;D3 02	..
	CALL	TIMER	;CD 22 01	..
	JP	J1	;C3 11 01	...

;サブルーチン

TIMER:	LD	D,0E7H	;16 E7	..
J2:	LD	E,05FH	;1E 5F	..
J3:	DEC	E	;1D	.
	JP	NZ,J3	;C2 26 01	..
	DEC	D	;15	.
	JP	NZ,J2	;C2 24 01	..
	RET		;C9	.

図3-10 (a) FTEST1.MAC

TIMER

J2

RRRRRRRRRRRRRRRRRRRRRRRR

RRRRRRRRRRRRRRRRRRRRRRRRRRRRRR

RRRRRRRRRRRRRRRRRRRRRRRRRRRRRR

RRRRRRRRRRRRRRRRRRRRRRRRRRRRRR

RRRRRRRRRRRRRRRRRRRRRRRRRRRR

RRRRRRRRRRRRRRRRRRRRRRRRRRRRRR

参 考 文 献

1. MIFES Ver 5.0 ユーザーズガイド メガソフト
2. Z80系 クロスアセンブラ ZASM 取扱説明書 マイクロ・オーグ
3. XA80 取扱説明書 システム・ロード
4. Z80 マイコンプログラミング実習 太平洋工業 日刊工業新聞
5. AV-Link ユーザーズマニュアル アノール
6. ディスアセンブラ ZZ 取扱説明書 マイクロ・オーグ